server may be implemented as a Sun Sparc/Solaris server. The Transducer Design Advisor (TDA) software 86, the Microsoft Excel program 88 and various DOE software tools 90 reside on the user's PC 80. The DOE tools 90 are incorporated as extensions to the Excel program. The remote server 82 comprises the following software components: a core simulation engine 92; a FilesetServer program 94 which handles transactions between the Transducer Design Advisor 86 and the remote server software; an itsucalc program 96 which runs a single (non-DOE) simulation specified by a master parameters file (*.EXIC) and returns a binary results file (*.SIMD) containing either contour maps or an image, depending on command line options; a doeexe program 98 which runs simulations in the DOE mode (most of the code is common between itsucalc and doeexe; only the outer layers differ); and an analysis server 100, which is a computer program that facilitates analysis integration by providing communication links between the simulator and Excel, i.e., the analysis server translates standard Set, Get and Execute commands coming from Excel into a language that the simulator can understand. As used hereinafter, the term "simulator" refers to the doeexe program 98 in conjunction with the core simulation engine 92.

[0047] In addition to creating the files that define a DOE analysis, the Transducer Design Advisor also has some auxiliary functions to make the user's life easier. First, the Transducer Design Advisor will copy all of the files for an analysis over to the remote host that the user wants to run the simulation on, make the necessary edits to these files for the particular file system that the remote host has, and install them in the right directories for access by the DOE tools 90 and the analysis server 100. Second, the Transducer Design Advisor will also let the user run a simulation with a specific set of parameters, and then present the results in a variety of graphical formats. Performing single runs is a good way to let the user gain insight into behavior of the simulator, and the graphical output presents a wealth of information that is not captured in its entirety in the CTQs. Third, the Transducer Design Advisor will run a simulation which includes a simple

"patient" model and display the resulting image. The user can directly see sidelobes, speckle pattern, and other image features that are marks of image quality. Seeing an image can sometimes be easier and more direct than just looking at numeric CTQs.

[0048] All of the auxiliary functions for the Transducer Design Advisor 86 are implemented as the client half of a client-server pair, with the server functions being implemented on the remote host 82 by the FilesetServer process 94. This is the server process that a Transducer Design Advisor client connects to for the purposes of: (a) uploading a simulation fileset to a server; (b) uploading a phantom specification file to a server; and (c) performing a single (non-DOE) simulation run. The server-side software components are capable of handling any number of simultaneous client connections, but for the sake of simplicity, this disclosure will only discuss the example of a single client. The FilesetServer program 94 preferably resides on the remote server 92 and is written in Java 2 in the preferred embodiment. The main network protocol used between the Transducer Design Advisor 86 and the FilesetServer program 94 is TCP/IP.

[0049] In accordance with the preferred embodiment of the invention, the Transducer Design Advisor 86 is a Java application which helps create and modify the files needed to use an ultrasound simulator with a Design of Experiment (DOE) toolset. Most users will run the Transducer Design Advisor on their personal computers (as seen in FIG. 4), but it can also run on Unix systems. The Transducer Design Advisor is written to be easy to use. Its user interface is similar to the familiar Windows Wizards. It includes extensive on-line help to get new users up and running quickly, and also supports quick navigation methods to help experienced users get their work done faster.

[0050] When the ultrasound simulator is used with the DOE tools, there are five text files that must be created before running any simulation. Two of these files are required by the DOE software, and three

are required by the ultrasound simulator. The Transducer Design Advisor creates these files based on a series of questions that the user answers as he/she moves through the Transducer Design Advisor windows. If the user is setting up a brand new simulation, the Transducer Design Advisor will walk the user through the process of creating these files from scratch. If the user is updating some files from a previous simulation, where the user only needs to tweak a few parameters, the Transducer Design Advisor will let the user edit these files efficiently and with much less chance of error than if the user used a text editor.

[0051] The two files required by the DOE software are the File Wrapper file ("*.fileWrapper") and the I/O Template file ("*.tplt"). These files allow the ultrasound simulator to talk to analysis server 100 and to pass data from the user's Excel spreadsheet 88 to the simulation and back again. For simple analysis codes, it is quite easy to write the FileWrapper and Template files by hand, and the user may never have to modify them once they are written. For the ultrasound simulator, however, the user needs to tailor these two files for a particular simulation to be run, and the set of input and output variables that the user will use may differ greatly from simulation to simulation. This happens because the largest realistic DOE experiment that one can run has only a handful of input variables, but the ultrasound simulator itself uses several hundred input parameters. The relevant output variables will also change depending on the kind of transducer being simulated.

[0052] The three files required by the ultrasound simulator for its internal use help it to expand the DOE input variables into the full parameter set that it needs to run. These three files are the Master EXIC file ("*_master.exic"), the Variable Definition file ("*.vdf"), and the CTQ Policy file ("*.pol"). The Master EXIC file contains a prototype of the full parameter file that the simulator uses. The Variable Definition File defines how the DOE input variables map into the EXIC structure. The CTQ Policy file defines the relative weights for different CTQs at different depths, so that performance